

A<sup>2</sup> field **43** as the number of instructions succeeding the execution control instruction, should be nullified. Suppose the nullification controller **30** has determined that the number **N** of instructions should be nullified since the execution condition **EC** is not met, the controller **30** asserts a nullification signal **NUL** to be supplied to the ALU **21**. In that case, even if the number **N** of instructions following the execution control instruction have already been executed to a midway point of the pipeline, these instructions are nullified in the ALU **21**. Accordingly, the same results are attained as if NOP (no operation) instructions had been executed instead of these instructions. Alternatively, if the nullification controller **30** has determined that the number **N** of succeeding instructions should not be nullified since the execution condition **EC** is met, then the nullification signal **NUL** is not asserted. Thus, the number **N** of instructions succeeding the execution control instruction are enabled and executed by the ALU **21**.

#### IN THE CLAIMS:

Claims 1-3, 5, 7 and 9-12 have been amended, and new claim 13 have been added as follows:

- A<sup>3</sup> h<sup>99</sup> C 1. A system for controlling the execution of a program in an information processor for processing instructions by pipelining, the system comprising:  
means for providing a set of instructions including an execution control instruction, the execution control instruction containing a condition field and an instruction-specifying field, the condition field specifying, in binary code, an execution

condition, the instruction-specifying field defining, in binary code, the number of instructions to be executed only if said execution condition is satisfied;

A3 means for deciding, based on the results of operations performed in response to one or more instructions preceding the execution control instruction in the instruction set provided, whether or not the execution condition that has been specified by the condition field of the execution control instruction is satisfied; and

means for determining based on the outcome of the decision whether or not said number of instructions, which number has been defined by the instruction-specifying field as the number of instructions succeeding the execution control instruction, should be nullified.

2. A method for controlling the execution of a program in an information processor for processing instructions by pipelining, the method comprising the steps of:

a) providing a set of instructions including an execution control instruction, the execution control instruction containing a condition field and an instruction-specifying field, the condition field specifying, in binary code, an execution condition, the instruction-specifying field defining, in binary code, the number of instructions to be executed only if said execution condition is satisfied;

b) deciding, based on the results of operations performed in response to one or more instructions preceding the execution control instruction in the instruction set provided, whether or not the execution condition that has been specified by the condition field of the execution control instruction is satisfied; and

A3 c) determining based on the outcome of the decision step b) whether or not said number of instructions, which number has been defined by the instruction-specifying field as the number of instructions succeeding the execution control instruction, should be nullified.

3. The method of claim 2, wherein the condition field is a single field for specifying the execution condition, and

wherein the instruction-specifying field is a single field for defining the instruction number, and

wherein the step c) comprises the sub-step of regarding said number of instructions, which number has been defined by the instruction-specifying field as the number of instructions succeeding the execution control instruction, as instructions to be executed only if said execution condition is satisfied, and nullifying the conditionally executable instructions if the execution condition that has been specified by the condition field is not satisfied.

A4 5. The method of claim 2, wherein the condition field is a single field for specifying the execution condition, and

wherein the instruction-specifying field is a single field for defining the instruction number, and

wherein the step c) comprises the sub-steps of:

AM

regarding said number of instructions, which number has been defined by the instruction-specifying field as a number of instructions succeeding the execution control instruction, as a first set of conditionally executable instructions, and nullifying the first set of conditionally executable instructions if the execution condition that has been specified by the condition field is not satisfied; and

regarding said number of instructions, which number has been defined by the instruction-specifying field as the number of instructions succeeding the first set of conditionally executable instructions, as a second set of conditionally executable instructions, and nullifying the second set of conditionally executable instructions if the execution condition that has been specified by the condition field is satisfied.

AS

7. The method of claim 2, wherein the condition field is a single field for specifying the execution condition, and

wherein the instruction-specifying field contains first and second instruction-specifying sub-fields, which respectively define, in binary code, first and second numbers of instructions to be executed only conditionally, and

wherein the step c) comprises the sub-steps of:

regarding the first number of instructions, which number has been defined by the first instruction-specifying sub-field as the number of instructions succeeding the execution control instruction, as a first set of conditionally executable instructions, and nullifying the first set of conditionally executable instructions if the execution condition specified by the condition field is not satisfied; and

A5

regarding the second number of instructions, which number has been defined by the second instruction-specifying sub-field as the number of instructions succeeding the first set of conditionally executable instructions, as a second set of conditionally executable instructions, and nullifying the second set of conditionally executable instructions if the execution condition specified by the condition field is satisfied.

A6

9. The method of claim 2, wherein the condition field includes multiple condition sub-fields, each specifying a single associated execution condition, and wherein the instruction-specifying field is a single field for defining the instruction number, and

wherein the step c) comprises a plurality of sub-steps, and wherein in each said sub-step, said number of instructions, which number has been defined by the instruction-specifying field as the number of instructions succeeding the execution control instruction, are regarded as conditionally executable instructions, and, if the execution condition specified by an associated one of the condition sub-fields is not satisfied, the conditionally executable instructions at a location corresponding to the execution condition specified are nullified.

10. The method of claim 9, wherein the step c) comprises a plurality of sub-steps,

A6  
wherein in each said sub-step, if the execution condition specified by an associated one of the condition sub-fields is satisfied, the conditionally executable instructions at a location corresponding to the execution condition specified are executed.

11. The method of claim 2, wherein the condition field includes multiple condition sub-fields, each specifying a single execution condition, and wherein the instruction-specifying field includes multiple instruction-specifying sub-fields corresponding to the respective condition sub-fields, each said instruction-specifying sub-field defining an associated instruction number in binary code, and wherein the step c) comprises a plurality of sub-steps, and wherein in each said sub-step, said number of instructions, which number has been defined by an associated one of the instruction-specifying sub-fields as the number of instructions succeeding the execution control instruction, are regarded as conditionally executable instructions, and, if the execution condition specified by associated one of the condition sub-fields is not satisfied, the conditionally executable instructions at a location corresponding to the execution condition specified are nullified.

12. The method of claim 11, wherein the step c) comprises a plurality of sub-steps, wherein in each said sub-step, if the execution condition specified by an associated one of the condition sub-fields is satisfied, the conditionally executable

A6 instructions at the location corresponding to the execution condition specified are executed.

---

New claim 13 have been added as follows:

---

A7 13. A method for controlling the execution of a program in an information processor for processing instructions by pipelining, the method comprising the steps of:

- a) providing a set of instructions including an execution control instruction, the execution control instruction containing a condition field specifying an execution condition in binary code;
  - b) deciding, based on the results of operations performed in response to one or more instructions preceding the execution control instruction in the instruction set provided, whether or not the execution condition that has been specified by the condition field of the execution control instruction is satisfied; and
  - c) nullifying one or more instructions succeeding the execution control instruction if the execution condition that has been specified by the condition field of the execution control instruction is not satisfied.
-